

```

#R CMD INSTALL \\Aa.ad.epa.gov\ord\RTP\Users\E-J\jwambaug\NETMyD~1\Resear~1\vLiverPBPK

#setwd("\\\\Aa/ord/RTP/USers/E-J/jwambaug/NET MyDocuments/Research Projects/vLiverPBPK")

Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^4))
Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^3),CAS.end=37)
Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^3),CAS.start=38,CAS.end=74)

Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))
Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,random.suppress=0.1,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))

Make.Wetmore.Plot(Wetmore,predictions.lower.col=NULL,cex.compound=0.15,ylab="AC50 (uM)",dose.lower.col="AC50",dose.order.col="Lower.Oral.Equivalent.mg.kg.",predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-5,10^4))
Make.Wetmore.Plot(Wetmore,predictions.lower.col=NULL,cex.compound=0.15,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-5,10^4))

Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))

UVPKM <-
function(model,fixed.params,which.quantile=0.95,CV.params=NULL,unif.params=NULL,samples=1000)
{
  sample.vec <- rep(NA,samples)
  for (this.sample in 1:samples)
  {
    these.params <- fixed.params
    for (this.param in names(CV.params))
    {
      these.params[[this.param]] <-
rnorm(1,mean=CV.params[[this.param]]$mean,sd=CV.params[[this.param]]$mean*CV.params[[this.param]]$CV)
    }
    for (this.param in names(unif.params))
    {
      these.params[[this.param]] <-
runif(1,min=unif.params[[this.param]]$min,max=unif.params[[this.param]]$max)
    }
    sample.vec[this.sample] <- call(model,these.params)
  }

  return(quantile(sample.vec,which.quantile))
}

Calc_1comp_Css <- function(chem.name=NULL,chem.CAS=NULL,daily.dose=1,which.quantile=0.95)

```

```

{

Wetmore_Css(chem.CAS="94-82-6")
Wetmore_Css(chem.CAS="94-82-6",which.quantile=0.5)

calc_Css(chem.CAS="94-82-6")

Wetmore_Oral_Equiv(4,chem.CAS="94-82-6")
Wetmore_Oral_Equiv(6,chem.CAS="94-82-6")
Wetmore_Oral_Equiv(6,chem.CAS="94-82-6",species="Rat")
Wetmore_Oral_Equiv(6,chem.CAS="94-75-7",species="Rat",which.quantile=0.5)

Wetmore.human.values <- NULL
vLiver.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    Wetmore.human.values[this.CAS] <- Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5)
    vLiver.human.values[this.CAS] <- calc_Css(parameterize_SteadyState(this.CAS))
  }

plot(vLiver.human.values,Wetmore.human.values,log="xy")
matplot(c(10^-3,10^2),c(10^-3,10^2),lty=2,add=T,type="l")

these.params <- parameterize_SteadyState("34256-82-1")
vary.params <- list(BW=list(mean=70,CV=0.2))
vary.params[["liver.volume.per.kgBW"]]<-list(mean=0.02448,CV=0.2)
vary.params[["QGFRc"]]<-list(mean=0.108,CV=0.2)
vary.params[["Qhc"]]<-list(mean=2.228571,CV=0.2)
vary.params[["million.cells.per.gliver"]]<-list(mean=110,CV=0.2)

UVPKM("calc_Css",these.params,CV.params=vary.params)

Wetmore.human.values <- NULL
vLiver.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    Wetmore.human.values[this.CAS] <- Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5)
    these.params <- parameterize_SteadyState(this.CAS)
    vLiver.human.values[this.CAS] <-
UVPKM("calc_Css",these.params,CV.params=vary.params,which.quantile=0.5)
  }

plot(vLiver.human.values,Wetmore.human.values,log="xy")
matplot(c(10^-3,10^2),c(10^-3,10^2),lty=2,add=T,type="l")

```

```

Wetmore.human.values <- NULL
vLiver.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    Wetmore.human.values[this.CAS] <- Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.95)
    these.params <- parameterize_SteadyState(this.CAS)
    vLiver.human.values[this.CAS] <-
      UVPKM("calc_Css",these.params,CV.params=vary.params,which.quantile=0.95)
  }
  plot(vLiver.human.values,Wetmore.human.values,log="xy")
  matplot(c(10^-3,10^2),c(10^-3,10^2),lty=2,add=T,type="l")

```

```

Wetmore.human.values <- NULL
vLiver.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    Wetmore.human.values[[this.CAS]] <-
      c(Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.05),Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5),Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.95))
    these.params <- parameterize_SteadyState(this.CAS)
    vLiver.human.values[[this.CAS]] <-
      UVPKM("calc_Css",these.params,CV.params=vary.params,which.quantile=c(0.05,0.5,0.95))
  }

```

```

Wetmore.rat.values <- NULL
vLiver.rat.values <- NULL
for (this.CAS in get_Wetmore_CAS("Rat"))
  if (this.CAS %in% get_Css_CAS("Rat"))
  {
    Wetmore.rat.values[[this.CAS]] <-
      Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5,species="Rat")
    these.params <- parameterize_SteadyState(this.CAS,species="Rat")
    vLiver.rat.values[[this.CAS]] <-
      UVPKM("calc_Css",these.params,CV.params=vary.params,which.quantile=0.5)
  }

```

```

plot(10^-6,10^-6,xlim=c(10^-3,10^3),ylim=c(10^-3,10^3),log="xy",xlab="Css Predicted",ylab="Css
Wetmore et al. (2012,2013)")
for (this.CAS in names(Wetmore.human.values))
{
  matplot(vLiver.human.values[[this.CAS]][1],Wetmore.human.values[[this.CAS]][1],pch=0,add=T)
  matplot(vLiver.human.values[[this.CAS]][2],Wetmore.human.values[[this.CAS]][2],pch=1,add=T)
  matplot(vLiver.human.values[[this.CAS]][3],Wetmore.human.values[[this.CAS]][3],pch=2,add=T)
}
for (this.CAS in names(Wetmore.rat.values))
{
  matplot(vLiver.rat.values[[this.CAS]][1],Wetmore.rat.values[[this.CAS]][1],pch=16,add=T)
}
```

```
}

matplot(c(10^-4,10^4),c(10^-4,10^4),lty=2,add=T,type="l")
matplot(90,10^-1,pch=0,add=T)
matplot(90,10^-1.25,pch=1,add=T)
matplot(90,10^-1.5,pch=2,add=T)
text(100,10^-1,"Lower 95th",pos=4)
text(100,10^-1.25,"Median",pos=4)
text(100,10^-1.5,"Upper 5th",pos=4)
```